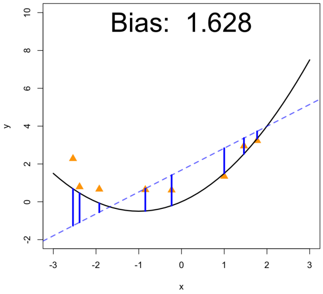
Introduction to Machine Learning

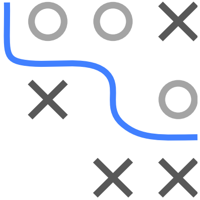
Advanced Risk Minimization
Bias-Variance 1:
Bias-Variance Decomposition



Learning goals

- Decompose GE of learner into
  - bias of learner
  - variance of learner
  - inherent noise of data
- Simulation study demo
- Capacity and overfitting

BIAS-VARIANCE DECOMPOSITION



- Generalization error of learner $\mathcal{I}$: Expected error of model $\mathcal{I}(\mathcal{D}_n) = \hat{f}_{\mathcal{D}_n}$, trained on set of size n, evaled on fresh test sample

$$GE_n\left(\mathcal{I}\right) = \mathbb{E}_{\mathcal{D}_n \sim \mathbb{P}_{xy}^n, (\mathbf{x}, y) \sim \mathbb{P}_{xy}} \left[L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) \right] = \mathbb{E}_{\mathcal{D}_n, xy} \left[L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) \right]$$

- $\mathbb{E}$ taken over all train sets **and** independent test sample. Could also frame this as expected risk (expectation over $\mathcal{D}_n$)

$$GE_n\left(\mathcal{I}\right) = \mathbb{E}_{\mathcal{D}_n} \mathbb{E}_{xy} \left[L\left(y, \hat{f}_{\mathcal{D}_n}(\mathbf{x})\right) \right] = \mathbb{E}_{\mathcal{D}_n} \left[\mathcal{R}(\hat{f}_{\mathcal{D}_n}) \right]$$

- For L2 loss, can additively decompose $GE_n\left(\mathcal{I}\right)$ into 3 components
- Assume data is generated by

$$y = f_{\text{true}}(\mathbf{x}) + \epsilon$$

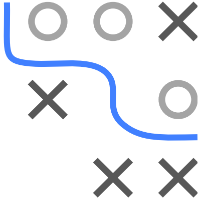
with 0-mean homoskedastic error $\epsilon \sim (0, \sigma^2)$; independent of $\mathbf{x}$

- Similar decomps exist for other losses expressable as Bregman divergences (e.g. log-loss). One exception is $0/1$ ▸ Brown and Ali 2024

BIAS-VARIANCE DECOMPOSITION

$$GE_n(\mathcal{I}) =$$

$$\underbrace{\sigma^2}_{\text{Var. of } \epsilon} + \mathbb{E}_x \underbrace{\left[\text{Var}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x}) \mid \mathbf{x}\right)\right]}_{\text{Variance of learner at } \mathbf{x}} + \mathbb{E}_x \underbrace{\left[\left(\left(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}\left(\hat{f}_{\mathcal{D}_n}(\mathbf{x})\right)\right)^2 \mid \mathbf{x}\right)\right]}_{\text{Squared bias of learner at } \mathbf{x}}$$



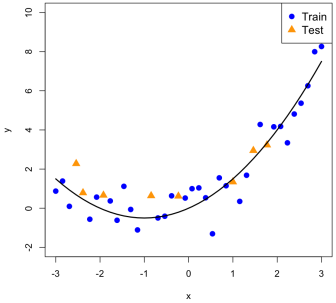
1. First: variance of "pure" **noise** ϵ; aka Bayes, intrinsic or irreducible error; whatever we we do, will never be better
2. Second: how much $\hat{f}_{\mathcal{D}_n}(\mathbf{x})$ **fluctuates** at test $\mathbf{x}$ if we vary training data, averaged over feature space; = learner's tendency to learn random things irrespective of real signal (overfitting)
3. Third: how "off" are we on average at test locations (underfitting); uses "average model integrated out over all $\mathcal{D}_n$; models with high capacity have low **bias** and vice versa

SIMULATION EXAMPLE

- True model:

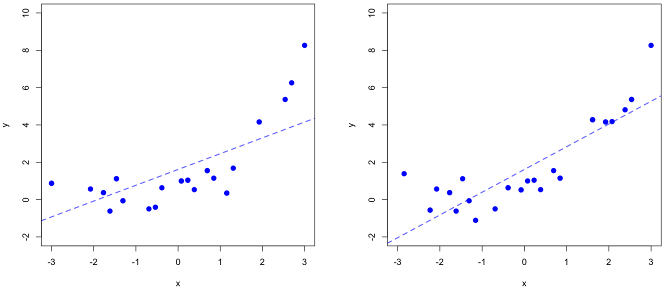
$$y = x + \frac{x^2}{2} + \epsilon \qquad \epsilon \sim N(0, 1)$$

- Split in train and test sets



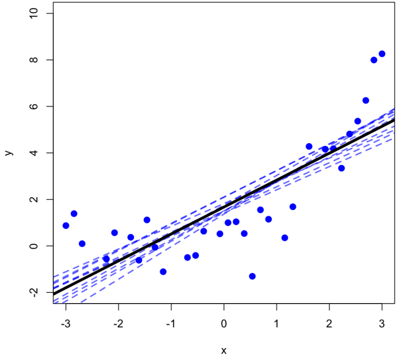
SIMULATION EXAMPLE

- Let's estimate bias and variance via bootstrapping
- (Could have also used Monte Carlo integration of the above quantities, BS slightly easier to visually explain)
- First, train several (low capacity) LMs
- These are the $\hat{f}_{\mathcal{D}_n}(\mathbf{x})$, seen as a RV, based on the random data $\mathcal{D}_n$



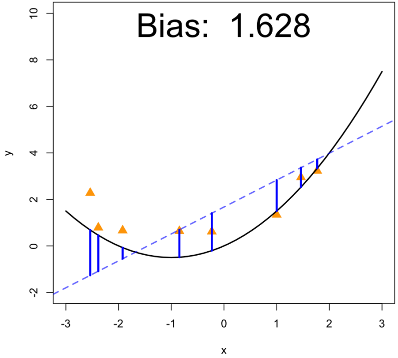
AVERAGE MODEL

- Average model over different training datasets
- This is $\mathbb{E}_{\mathcal{D}_n}[\hat{f}_{\mathcal{D}_n}(\mathbf{x})]$ in the decomp



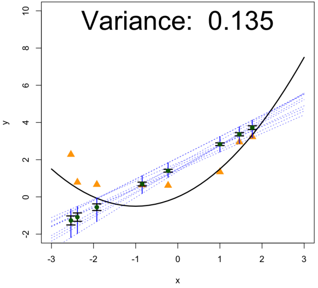
SQUARED BIAS COMPUTATION / ESTIMATION

- Compute sq. diff. between avg. and true model at each test x
- Then average over all test points
- This is $\mathbb{E}_x[(f_{\text{true}}(\mathbf{x}) - \mathbb{E}_{\mathcal{D}_n}(\hat{f}_{\mathcal{D}_n}(\mathbf{x})))^2 \mid \mathbf{x})]$



VARIANCE COMPUTATION

- Compute variance of model predictions at each test x
- Then average over all test points
- This is $\mathbb{E}_x[\text{Var}_{\mathcal{D}_n}(\hat{f}_{\mathcal{D}_n}(\mathbf{x}) \mid \mathbf{x})]$

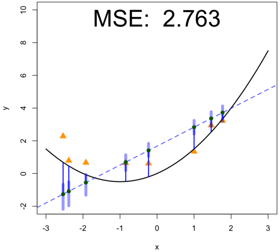


- Here, we know data variance $\sigma^2 = 1$;
  could also estimate it from residuals

DECOMP RESULT AND COMPARISON WITH MSE

- Decomp result; here bias is largest:

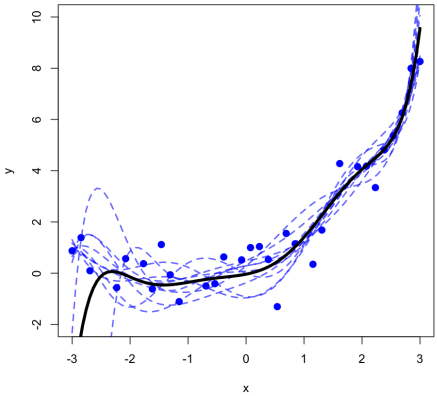
$$GE_n(\mathcal{I}) \approx 1 + 1.628 + 0.135 = 2.763$$

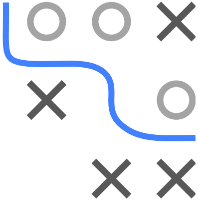


- Regular MSE: For each model, compute MSE on test set
- Then we average these MSEs over all models
- Result = 2.72; checks out;
  better if we avg. over more models and test points
- In general: Error quite high as we underfitted

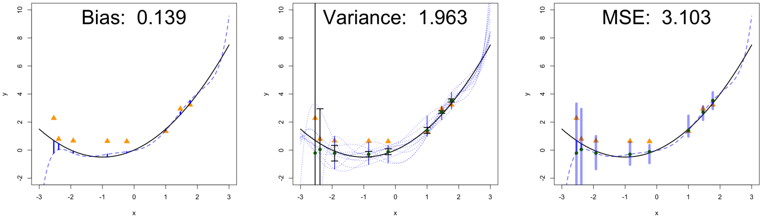
HIGHER COMPLEXITY LEARNER

- Same procedure, but using a high-degree polynomial ($d = 7$)





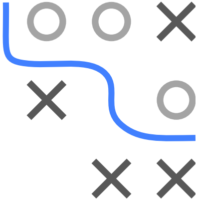
HIGHER COMPLEXITY LEARNER



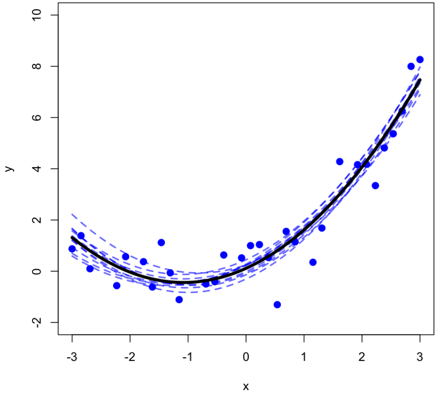
$$GE_n(\mathcal{I}) \approx 1 + 0.139 + 1.963 \approx 3.103$$

- GE higher than before, although hypo space now contains f_{true}
- Bias is lower, and variance higher
- Higher capacity learner overfits (here).
  We also do not regularize, that would be better
- NB: There is an “edge effect” on LHS, Runge effect,
  leads to higher bias as “artifact” here (ignore this)

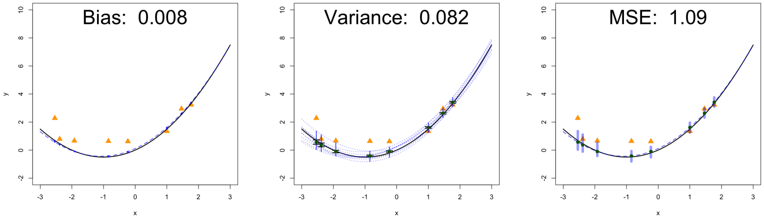
HIGHER COMPLEXITY LEARNER



- What happens if we use a model with the same complexity as the true model (quadratic polynomial)?



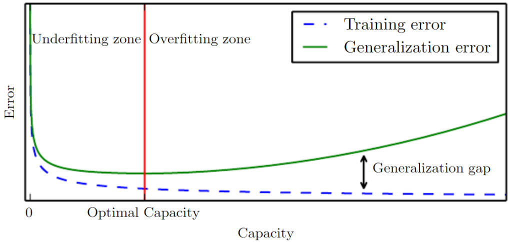
HIGHER COMPLEXITY LEARNER



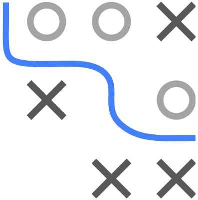
$$GE_n\left(\mathcal{I}\right) \approx 1 + 0.008 + 0.082 = 1.09$$

- Naturally: better result
- Low bias, low variance
- Bias should not be that much lower than high degree polynomial; but see comment there
- In any case, variance of the data is lower bound

CAPACITY AND OVERFITTING

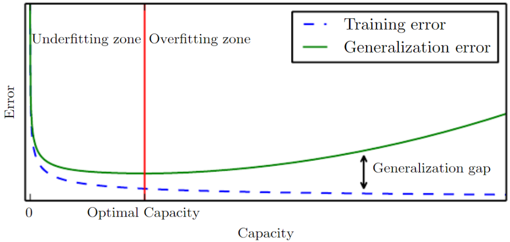


Credit: Ian Goodfellow

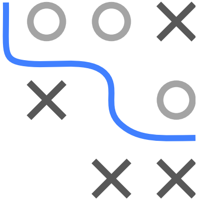


- Performance of a learner depends on its ability to
  1. **fit** the training data well
  2. **generalize** to new data
- Failure of the first point is called **underfitting**
- Failure of the second point is called **overfitting**

CAPACITY AND OVERFITTING



Credit: Ian Goodfellow



- The tendency of a learner to underfit/overfit is a function of its capacity, determined by the type of hypotheses it can learn
- Usually: high capacity $\rightarrow$ low bias $\rightarrow$ better fit on train
- But: high capacity $\rightarrow$ high variance $\rightarrow$ high chance of overfitting
- For such models, regularization (discussed later) is essential
- Even for correctly specified models, the generalization error is lower-bounded by the irreducible noise σ^2